Compression and Recovery of Distributed Random Signals

Alex Grant, Anatoli Torokhti, and Pablo Soto-Quiros

Abstract

We consider the case when a set of spatially distributed sensors $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$ make local observations, $\mathbf{y}_1, \ldots, \mathbf{y}_p$, which are noisy versions of a signal of interest, $\mathbf{x}$. Each sensor $\mathcal{Q}_j$ transmits compressed information $\mathbf{u}_j$ about its measurements to the fusion center which should recover the original signal within a prescribed accuracy. Such an information processing relates to a wireless sensor network (WSN) scenario. The key problem is to find models of the sensors and fusion center so that they will be optimal in the sense of minimization of the associated error under a certain criterion, such as the mean square error (MSE). We determine the models from the technique which is a combination of the maximum block improvement (MBI) method [1], [2] and the generic Karhunen-Loève transform (KLT) [3] (based on the work in [4], [5]). Therefore, the proposed method unites the merits of both techniques [1], [2] and [3], [4], [5]. As a result, our approach provides, in particular, the minimal MSE at each step of the version of the MBI method we use. The WSN model is represented in the form called the multi-compressor KLT-MBI transform. The multi-compressor KLT-MBI is given in terms of pseudo-inverse matrices and, therefore, it is numerically stable and always exists. In other words, the proposed WSN model provides compression, de-noising and reconstruction of distributed signals for the cases when known methods either are not applicable (because of singularity of associated matrices) or produce larger associated errors. Error analysis is provided.

Index Terms

Alex Grant was with the Institute for Telecommunications Research, University of South Australia, SA 5095, Australia.

Anatoli Torokhti is with the Centre for Industrial and Applied Mathematics, University of South Australia, SA 5095, Australia (e-mail: anatoli.torokhti@unisa.edu.au).

Pablo Soto-Quiros is with the Centre for Industrial and Applied Mathematics, University of South Australia, SA 5095, Australia and the Instituto Tecnologico de Costa Rica, Apdo. 159-7050, Cartago, Costa Rica (e-mail: juan.soto-quiros@mymail.unisa.edu.au).

Manuscript received XXXX XX, 2015; revised XXXX XX, 2015.

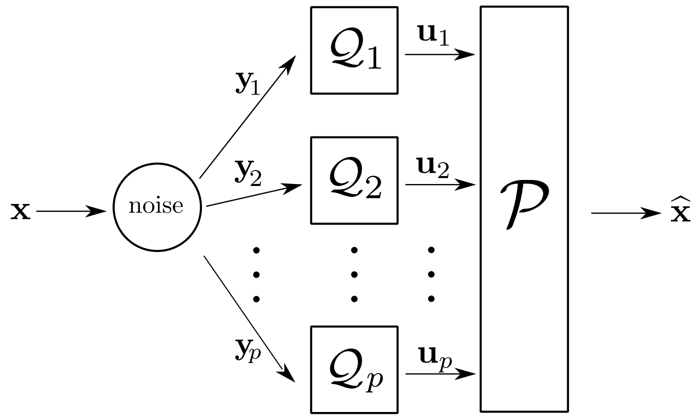


Fig. 1. Block diagram of the WSN. Here, $\widehat{\mathbf{x}}$ is an estimation of $\mathbf{x}$.

Karhunen-Loève transform, singular value decomposition, rank-reduced matrix approximation.

I. Introduction

A. Motivation

We seek to find effective numerical algorithms for an information processing scenario that involves a set of spatially distributed sensors, $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$, and a fusion center, $\mathcal{P}$. The sensors make local observations, $\mathbf{y}_1, \ldots, \mathbf{y}_p$, which are noisy versions of a signal of interest, $\mathbf{x}$. Each sensor $\mathcal{Q}_j$ transmits compressed information $\mathbf{u}_j$ about its measurements to the fusion center which should recover the original signal within a prescribed accuracy. Such an information processing relates to a wireless sensor network (WSN) scenario. In the recent years, research and development on new and refined WSN techniques has increased at a remarkable rate (see, for example, [6], [7], [8], [9], [10], [11], [12]). This is, in particular, because of a multitude of WSN applications due to their low deployment and maintenance cost.

The key problem is to to find an effective way to *compress* and denoise each observation $\mathbf{y}_j$, where $j = 1, \ldots, p$, and then *reconstruct* all the compressed observations in the fusion center so that the reconstruction will be optimal in the sense of a minimization of the associated error under a certain criterion, such as the mean square error (MSE). A restriction is that the sensors cannot communicate with each other. Here, the term “compression” is treated in the same sense as in the known works on data compression (developed, for instance, in [13], [14], [15], [16]), i.e. we say that observed signal $\mathbf{y}_j$ with n_j components is compressed if it is represented as

signal $\mathbf{u}_j$ with r_j components where $r_j < n_j$, for $j = 1, \ldots, p$. That is "compression" refers to dimensionality reduction and not quantization which outputs bits for digital transmission. This is similar to the way considered, in particular, in [17].

B. *Known techniques*

It is known that in the nondisrtibuted setting (in the other words, in the case of a single sensor only) the MSE optimal solution is provided by the Karhunen-Loève transform (KLT) [13], [14], [18], [3]. Nevertheless, the classical KLT cannot be applied to the above WSN since the entire data vector $\mathbf{y} = [\mathbf{y}_1^T, \ldots, y_p^T]^T$ is not observed by each sensor. Therefore, several approaches to a determination of mathematical models for $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$ and $\mathcal{P}$ have been pursued. In particular, in the information-theoretic context, distributed compression has been considered in the landmark works of Slepian and Wolf [19], and Wyner and Ziv [20]. A transform-based approach to distributed compression and the subsequent signal recovery has been considered in [11], [17], [21], [22], [23], [24]. Intimate relations between these two perspectives have been shown in [25]. The methodology developed in [17], [21], [22], [23], [24] is based on the dimensionality reduction by linear projections. Such an approach has received considerable attention (see, for example, [6], [7], [8], [10], [26], [27]).

In this paper, we consider a further extension of the methodology studied in [17], [21], [22], [23], [24]. In particular, in [17], two approaches are considered. By the first approach, the fusion center model, $\mathcal{P}$, is given in the form $\mathcal{P} = [\mathcal{P}_1, \ldots, \mathcal{P}_p]$ where $\mathcal{P}_j$ is a 'block' of $\mathcal{P}$, for $j = 1, \ldots, p$, and then the original MSE cost function is represented as a sum of p decoupled MSE cost functions. Then approximations to $\mathcal{Q}_j$ and $\mathcal{P}_j$ are found as solution to each of the p associated MSE minimization (MSEM) problems. We observe that the original MSE cost function and the sum of p decoupled MSE cost functions are not equivalent. This is because the covariance matrix cannot be represented in a block diagonal form in the way presented in [17]. Some more related explanations can be found in [28]. Therefore, the first approach in [17] leads to the corresponding increase in the associated error. The second approach in [17] generalizes the results in [21], [22] in the following way. The original MSE cost function is represented, by re-grouping its terms, in the form similar to that presented by a summand in the decoupled MSE cost function. Then the minimum is seeking for each $\mathcal{P}_j\mathcal{Q}_j$, for $j = 1, \ldots, p$, while other terms $\mathcal{P}_k\mathcal{Q}_k$, for $k = 1, \ldots, j-1, j+1, \ldots, p$, are assumed to be fixed. The minimizer follows from

the known result given in [29] (Theorem 10.2.4). While the original MSEM problem is stated for a simultaneous determination of $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$ and $\mathcal{P}$, the approach in [17] requires to solve p local MSEM problems which are not equivalent to the original problem. To combine solutions of those p local MSEM problems, approximations to each sensor model $\mathcal{Q}_j$ are determined from an iterative procedure. Values of $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$ for the initial iteration are chosen randomly.

The approach in [24] is based on the ideas similar to those in the earlier references [17], [21], [22], [23][1], i.e. on the replacement of the original MSEM problem with the $p+1$ unconstrained MSEM problems for separate determination of approximations to $\mathcal{Q}_j$, for each $j = 1, \ldots, p$, and then an approximation to $\mathcal{P}$. First, an approximation to each $\mathcal{Q}_j$, for $j = 1, \ldots, p$, is determined under assumption that other $p-1$ sensors are fixed. Then, on the basis of known approximations to $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$, an approximation of $\mathcal{P}$ is determined as the optimal Wiener filter. Those $p+1$ problems considered in [24] are not equivalent to the original problem. In [24], the involved signals are assumed to be zero-mean jointly Gaussian random vectors. Here, this restriction is not used.

The work in [21], [22], [23] can be considered as a particular case of [24].

The method in [11] is applied to the problem which is an approximation of the original problem. It implies an increase in the associated error compared to the method applied to the original problem. Further, it is applicable under certain restrictions imposed on observations and associated covariance matrices. In particular, in [11], the observations should be presented in the special form $\mathbf{y}_j = H_j\mathbf{x} + \mathbf{v}_j$, for $j = 1, \ldots, p$ (where H_j is a measurement matrix and $\mathbf{v}_j$ is noise), and the covariance matrix formed by the noise vector should be *block-diagonal* and invertible. It is not the case here.

C. *Differences from known methods. Novelty and Contribution*

The WSN models in [11], [17], [21], [22], [23], [24] are justified in terms of inverse matrices. It is known that in the cases when the matrices are close to singular this may lead to instability and significant increase in the associated error. Moreover, when the matrices are singular, the algorithms [17], [21], [22], [23], [24] may not be applicable. This observation is illustrated by Examples 2, 3 and 4 in Section VI below where the associated matrices are singular and the

[1]In particular, it generalizes work [23] to the case when the vectors of interest do not to be directly observed at the sensors.

method [17] is not applicable. Although in [24], for the case when a matrix is singular, it is proposed to replace its inverse by the pseudo-inverse, such a simple replacement does not follow from the justification of the model provided in [24]. As a result, a simple substitution of the pseudo-inverse matrices instead of the inverse matrices may lead to the numerical instability as it is shown, in particular, in Example 4 in Section VI. In this regard, we also refer to references [3], [4], [30] where the case of rank-constrained data compression in terms of the pseudo-inverse matrices is studied.

Thus, methods in [17], [21], [22], [23], [24], [11] are *justified* only for full rank matrices used in the associated models. This is not the case here. On the basis of the methodology developed in [3], [4], [30], our method is rigorously justified for models with matrices of degenerate ranks (Sections III-A and VIII).

Further, the second approach in [17] is, in fact, the block coordinate descent (BCD) method [31]. The BCD method converges (to a local minimum) under the assumptions that the objective function and the space of optimization need to be convex or the minimum of the objective function is *uniquely* attended (see [31] and [32], p. 267). Those conditions are not satisfied for our method (as for the method in [17] as well). Unlike the BCD method [31] used in [17], the maximum block improvement (MBI) method [1], [2] avoids the requirements of the BCD method. The MBI method guaranties convergence to a coordinate-wise minimum point which is a local minimum of the objective function. Therefore, the approach proposed in this paper is based, in particular, on the idea of the MBI method.

As distinct from the technique in [11] our method is applied to the original minimization problem, not to an approximation of the original problem as in [11]. It allows us to avoid the increase in the associated error. We also do not impose any of the restrictions on our method as those in [11] mentioned in Section I-B above. In particular, we do not assume that the covariance matrix formed by the noise vector should be block-diagonal.

The methods in [17], [21], [22], [23], [24] have been developed under assumption that exact covariance matrices are known. A knowledge of exact covariance matrices might be a restrictive condition in some cases, in particular, when matrices are large. In this paper, these difficulties are mitigated to some extent.

Key advantages of the proposed method are as follows. The method represents a combination of the generic Karhunen-Loève transform (KLT) [3] (based on the work in [5]) and the MBI

method [1], [2]. Therefore, it unites the merits of both techniques [3], [5] and [1], [2]. As a result, our approach provides, in particular, the minimal MSE at each step of the version of the MBI method we use. The WSN model is represented in the form called the *multi-compressor KLT-MBI transform* and is based on the ideas which are different from those used in known methods [6], [7], [8], [10], [17], [21], [22], [23], [24], [26]. The multi-compressor KLT-MBI is given in terms of pseudo-inverse matrices and, therefore, it is numerically stable and always exists. In other words, the proposed WSN model provides compression, de-noising and reconstruction of distributed signals for the cases when known methods either are not applicable (because of singularity of associated matrices) or produce larger associated errors. This observation is supported, in particular, by results of simulations in Section VI.

D. Notation

Here, we provide some notation which is required to formalize the problem in the form presented in Section II-B below. Let us write (Ω, Σ, μ) for a probability space[2]. We denote by $\mathbf{x} \in L^2(\Omega, \mathbb{R}^m)$ the signal of interest[3] (a source signal to be estimated) represented as $\mathbf{x} = [\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}]^T$ where $\mathbf{x}^{(j)} \in L^2(\Omega, \mathbb{R})$, for $j = 1, \ldots, m$. Futher, $\mathbf{y}_1 \in L^2(\Omega, \mathbb{R}^{n_1}), \ldots, \mathbf{y}_p \in L^2(\Omega, \mathbb{R}^{n_p})$ are observations made by the sensors where $n_1 + \ldots + n_p = n$. In this regard, we write

$$\mathbf{y} = [\mathbf{y}_1^T, \ldots, \mathbf{y}_p^T]^T \quad \text{and} \quad \mathbf{y} = [\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(n)}]^T \tag{1}$$

where $\mathbf{y}^{(k)} \in L^2(\Omega, \mathbb{R})$, for $k = 1, \ldots, n$. We would like to emphasize *a difference between* $\mathbf{y}_j$ *and* $\mathbf{y}^{(k)}$*:* in (1), the observation $\mathbf{y}_j$, for $j = 1, \ldots, p$, is a 'piece' of random vector $\mathbf{y}$ (i.e. $\mathbf{y}_j$ is a random vector itself), and $\mathbf{y}^{(k)}$, for $k = 1, \ldots, n$ is an entry of y (i.e. $\mathbf{y}^{(k)}$ is a random variable).[4]

For $j = 1, \ldots, p$, let us define a sensor model $\mathcal{Q}_j : L^2(\Omega, \mathbb{R}^{n_j}) \to L^2(\Omega, \mathbb{R}^{r_j})$ by the relation

$$[\mathcal{Q}_j(\mathbf{y}_j)](\omega) = Q_j[\mathbf{y}_j(\omega)] \tag{2}$$

where $Q_j \in \mathbb{R}^{r_j \times n_j}$,

$$r = r_1 + \ldots + r_p, \quad \text{where} \quad r_j \leq n_j. \tag{3}$$

[2] $\Omega = \{\omega\}$ is the set of outcomes, Σ a $\sigma-$field of measurable subsets of Ω and $\mu : \Sigma \to [0, 1]$ an associated probability measure on Σ with $\mu(\Omega) = 1$.

[3] The space $L^2(\Omega, \mathbb{R}^m)$ has to be used because of the norm introduced below in (5).

[4] Therefore, $\mathbf{y}_j = [\mathbf{y}^{(n_0+n_1+\ldots+n_{j-1})}, \ldots, \mathbf{y}^{(n_1+\ldots+n_j)}]^T$ where $j = 1, \ldots, p$ and $n_0 = 1$.

Let us denote $\mathbf{u}_j = \mathcal{Q}_j(\mathbf{y}_j)$ and $\mathbf{u} = [\mathbf{u}_1^T, \ldots, \mathbf{u}_p^T]^T$, where for $j = 1, \ldots, p$, vector $\mathbf{u}_j \in L^2(\Omega, \mathbb{R}^{r_j})$ represents the compressed and filtered information vector transmitted by a jth sensor $\mathcal{Q}_j$ to the fusion center $\mathcal{P}$. A fusion center model is defined by $\mathcal{P} : L^2(\Omega, \mathbb{R}^r) \to L^2(\Omega, \mathbb{R}^m)$ so that

$$[\mathcal{P}(\mathbf{u})](\omega) = P[\mathbf{u}(\omega)], \tag{4}$$

where $P \in \mathbb{R}^{m\times r}$ and $\mathbf{u} \in L^2(\Omega, \mathbb{R}^r)$. To state the problem in the next section, we also denote

$$\mathbb{E}\left[\|\mathbf{x}(\omega)\|_2^2\right] := \|\mathbf{x}\|_\Omega^2 = \int_\Omega \|\mathbf{x}(\omega)\|_2^2 d\mu(\omega) < \infty, \tag{5}$$

where $\|\mathbf{x}(\omega)\|_2$ is the Euclidean norm of $\mathbf{x}(\omega) \in \mathbb{R}^m$. For convenience, we will use notation $\|\mathbf{x}\|_\Omega^2$, not $\mathbb{E}\left[\|\mathbf{x}(\omega)\|_2^2\right]$, to denote the norm in (5).

II. Formalization and Statements of the Problems

A. Preliminaries and Formalization of the Problem

For the WSN depicted in Fig. 1, the problem can be stated as follows: Find models of the sensors, $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$, and a model of the fusion center, $\mathcal{P}$, that provide

$$\min_{\mathcal{P}, \mathcal{Q}_1, \ldots, \mathcal{Q}_p} \left\| \mathbf{x} - \mathcal{P} \begin{bmatrix} \mathcal{Q}_1(\mathbf{y}_1) \\ \vdots \\ \mathcal{Q}_p(\mathbf{y}_p) \end{bmatrix} \right\|_\Omega^2 \tag{6}$$

under the assumption that $\mathcal{Q}_1, \ldots, \mathcal{Q}_p$ and $\mathcal{P}$ are given by (2) and (4), respectively. The model of the fusion center, $\mathcal{P}$, can be represented as $\mathcal{P} = [\mathcal{P}_1, \ldots, \mathcal{P}_p]$ where $\mathcal{P}_j : L^2(\Omega, \mathbb{R}^{r_j}) \to L^2(\Omega, \mathbb{R}^m)$, for $j = 1, \ldots, p$. Let us write

$$\begin{aligned} &\min_{\substack{\mathcal{P}_1, \ldots, \mathcal{P}_p, \\ \mathcal{Q}_1, \ldots, \mathcal{Q}_p}} \left\| \mathbf{x} - [\mathcal{P}_1, \ldots, \mathcal{P}_p] \begin{bmatrix} \mathcal{Q}_1(\mathbf{y}_1) \\ \vdots \\ \mathcal{Q}_p(\mathbf{y}_p) \end{bmatrix} \right\|_\Omega^2 \\ &\qquad = \min_{\substack{\mathcal{P}_1, \ldots, \mathcal{P}_p, \\ \mathcal{Q}_1, \ldots, \mathcal{Q}_p}} \|\mathbf{x} - [\mathcal{P}_1\mathcal{Q}_1(\mathbf{y}_1) + \ldots + \mathcal{P}_p\mathcal{Q}_p(\mathbf{y}_p)]\|_\Omega^2 \end{aligned} \tag{7}$$

where $\mathbf{y} = [\mathbf{y}_1^T, \ldots, \mathbf{y}_p^T]^T$.

B. Statement of the Problem

For $j = 1, \ldots, p$, let us denote $\mathcal{F}_j = \mathcal{P}_j \mathcal{Q}_j$. We also write $\mathcal{F} = [\mathcal{F}_1, \ldots, \mathcal{F}_p]$. Then the WSN model can be represented as

$$\mathcal{F}(\mathbf{y}) = \mathcal{P}\mathcal{Q}(\mathbf{y}) \quad \text{and} \quad \mathcal{F}(\mathbf{y}) = \sum_{j=1}^{p} \mathcal{F}_j(\mathbf{y}_j). \tag{8}$$

Denote by $\mathcal{R}(m, n, k)$ the variety of all $m \times n$ linear operators of rank at most k. For the sake of simplicity we sometimes will also write $\mathcal{R}_k$ instead of $\mathcal{R}(m, n, k)$. The problem in (7) can equivalently be reformulated as follows: Find $\mathcal{F}_1 \in \mathcal{R}(m, n_1, r_1)$, $\ldots$, $\mathcal{F}_p \in \mathcal{R}(m, n_p, r_p)$ that solve

$$\min_{\mathcal{F}_1 \in \mathcal{R}_{r_1}, \ldots, \mathcal{F}_p \in \mathcal{R}_{r_p}} \left\| \mathbf{x} - \sum_{j=1}^{p} \mathcal{F}_j(\mathbf{y}_j) \right\|_{\Omega}^{2}. \tag{9}$$

Recall that $r_j < n_j$, for $j = 1, \ldots, p$.

Further, to simplify the notation we will use the same symbol to denote an operator and the associated matrix. For example, we write Q_j to denote both the operator $\mathcal{Q}_j$ and matrix Q_j introduced in (2). Similarly, we write P_j to denote both operator $\mathcal{P}_j$ and matrix $P_j \in \mathbb{R}^{m \times r_j}$ introduced above, etc. In particular, by this reason, in (9) we will write F_j , not $\mathcal{F}_j$.

C. Assumptions

For $\mathbf{x}$ represented by $\mathbf{x} = [\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}]^T$ where $\mathbf{x}^{(j)} \in L^2(\Omega, \mathbb{R})$ we write

$$E[\mathbf{x}\mathbf{y}^T] = E_{xy} = \left\{ \langle \mathbf{x}^{(j)}, \mathbf{y}^{(k)} \rangle \right\}_{j,k=1}^{m,n} \in \mathbb{R}^{m \times n}, \tag{10}$$

where $\langle \mathbf{x}^{(j)} \mathbf{y}^{(k)} \rangle = \displaystyle\int_{\Omega} \mathbf{x}^{(j)}(\omega) \mathbf{y}^{(k)}(\omega) d\mu(\omega)$ and $\mathbf{y} = [\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(n)}]^T$.

The assumption used in the known methods [6], [7], [8], [10], [11], [17], [21], [22], [23], [24], [26], [27] is that covariance matrices E_{xy} and E_{yy} are known. At the same time, in many cases, it is difficult to know exact values of E_{xy} and E_{yy}. For instance, if it is assumed that signal $\mathbf{y}$ is a Gaussian random vector then the associated parameters ρ and σ^2 for matrix E_{yy} are still unknown (see [24] as an example). Therefore, we are mainly concerned with the case when estimates of E_{xy} and E_{yy} can be obtained. Methods of estimation of matrices E_{xy} and E_{yy} were studied in a number of papers (see, for example, [33], [34], [35], [36], [37], [38], [39], [40]) and it is not a subject of our work. In particular, samples of training signals taken

for some different random outcomes ω might be available. Some knowledge of the covariances can also come from specific data models. Then the aforementioned covariance matrices can be estimated.

In the following Section III, we provide solutions in terms of both matrices E_{xy}, E_{yy} and their estimates. The associated error analysis is given in Sections V and VIII.

D. Solution of Problem (9) for Single Sensor, i.e. for $p = 1$

Here, we recall the known result for the case of a single sensor, i.e. for $p = 1$ in (6) and (9), and provide some related explanations which will be extended in the sections that follow.

The Moore-Penrose generalized inverse for a matrix M is denoted by $M^\dagger$. We set $M^{1/2\dagger} = (M^{1/2})^\dagger$, where $M^{1/2}$ is a square root of M, i.e. $M = M^{1/2}M^{1/2}$. Then the known solution (see, for example [3]) of the particular case of problem (9), for $p = 1$, is given by

$$F_1 = \left[E_{xy_1}(E^\dagger_{y_1y_1})^{1/2}\right]_{r_1}(E^\dagger_{y_1y_1})^{1/2} + M_1[I - E^{1/2}_{y_1y_1}(E^\dagger_{y_1y_1})^{1/2}] \tag{11}$$

where symbol $[\cdot]_{r_1}$ denotes a truncated singular value decomposition (SVD) taken with r_1 first nonzero singular values and matrix M_1 is arbitrary. The expression (11) represents the Karhunen-Loève transform as given, for example, in [6].

In (11), in particular, $M_1 = \mathbb{O}$ where $\mathbb{O}$ is the zero matrix. Then Q_1 and P_1 that solve (6) follow from a decomposition of matrix $\left[E_{x,y_1}(E^\dagger_{y_1y_1})^{1/2}\right]_{r_1}(E^\dagger_{y_1y_1})^{1/2}$, based on the truncated SVD, in a product P_1Q_1 of $r_1 \times n$ matrix Q_1 and $m \times r_1$ matrix P_1 , respectively.

We will extend this argument in Section III below for a general case with more than one sensor.

III. Main Results

A. Greedy Approach to Solution of Problem (9), for $p = 1, 2, \ldots$

We wish to find $F_1, \ldots, F_p$ that provide a solution to the problem (9) for an arbitrary finite number of sensors in the WSN model, i.e. for $p = 1, 2, \ldots$ in (9). To this end, we need some more preliminaries which are given in Sections III-A1 and III-A2 that follow. The method itself and an associated algorithm are then represented in Section III-A3.

1) SVD, Orthogonal Projections and Matrix Approximation: Let the SVD of a matrix $C \in \mathbb{R}^{m\times s}$ be given by

$$C = U_C \Sigma_C V_C^T, \tag{12}$$

where $U_C \in \mathbb{R}^{m\times m}$ and $V_C \in \mathbb{R}^{s\times s}$ are unitary matrices, $\Sigma_C = \text{diag}(\sigma_1(C), \ldots, \sigma_{\min(m,s)}(C)) \in \mathbb{R}^{m\times s}$ is a generalized diagonal matrix, with the singular values $\sigma_1(C) \geq \sigma_2(C) \geq \ldots 0$ on the main diagonal. Let $U_C = [u_1\ u_2\ \ldots\ u_m]$ and $V_C = [v_1\ v_2\ \ldots\ v_s]$ be the representations of U and V in terms of their m and s columns, respectively. Let

$$L_C = \sum_{i=1}^{\text{rank } C} u_i u_i^T \in \mathbb{R}^{m\times m} \quad \text{and} \quad R_C = \sum_{i=1}^{\text{rank } C} v_i v_i^T \in \mathbb{R}^{s\times s} \tag{13}$$

be the orthogonal projections on the range of C and C^T, correspondingly. Define

$$C_r = [C]_r = \sum_{i=1}^{r} \sigma_i(C) u_i v_i^T = U_{C_r} \Sigma_{C_r} V_{C_r}^T \in \mathbb{R}^{m\times s} \tag{14}$$

for $r = 1, \ldots, \text{rank } C$, where

$$U_{C_r} = [u_1\ u_2\ \ldots\ u_r],\ \Sigma_{C_r} = \text{diag}(\sigma_1(C), \ldots, \sigma_r(C)) \text{ and } V_{C_r} = [v_1\ v_2\ \ldots\ v_r]. \tag{15}$$

For $r > \text{rank } C$, we write $C^{(r)} = C (= C_{\text{rank } C})$. For $1 \leq r < \text{rank } C$, the matrix $C^{(r)}$ is uniquely defined if and only if $\sigma_r(C) > \sigma_{r+1}(C)$.

Consider the problem: Given matrices S_j and G_j, for $j = 1, \ldots, p$, find matrix F_j that solves

$$\min_{F_j \in \mathbb{R}_{r_j}} \|S_j - F_j G_j\|^2, \quad \text{for } j = 1, \ldots, p. \tag{16}$$

The solution is given by Theorem 1 (which is a particular case of the results in [4], [5]) as follows.

Theorem 1: Let $K_j = M_j \left(I - L_{G_j}\right)$ where M_j is an arbitrary matrix. The matrix F_j given by

$$F_j = \left[S_j R_{G_j}\right]_{r_j} G_j^\dagger (I + K_j), \quad \text{for } j = 1, \ldots, p, \tag{17}$$

is a minimizing matrix for the minimal problem (16). Here, R_{G_j} and $[\cdot]_{r_j}$ are defined similarly to (13) and (14), respectively. Any minimizing F_j has the above form if and only if either

$$r_j \geq \text{rank } (S_j R_{G_j}) \tag{18}$$

or

$$1 \leq r_j < \text{rank } (S_j R_{G_j}) \quad \text{and } \sigma_{r_j}(S_j R_{G_j}) > \sigma_{r_j+1}(S_j R_{G_j}), \tag{19}$$

where $\sigma_{r_j}(S_j R_{G_j})$ is a singular value in the SVD for matrix $S_j R_{G_j}$.

Proof: The proof follows from [4], [5]. ■ ■

We note that the arbitrary matrix M_j implies non-uniqueness of solution (17) of problem (16).

2) Reduction of Problem (9) to Equivalent Form: We denote $F = [F_1, \ldots, F_p]$, where $F \in \mathbb{R}^{m\times n}$ and $F_j \in \mathbb{R}^{m\times n_j}$, for all $j = 1, \ldots, p$, and write $\|\cdot\|$ for the Frobenius norm. Then

$$\begin{aligned}\left\|\mathbf{x} - \sum_{j=1}^{p} F_j(\mathbf{y}_j)\right\|_{\Omega}^2 &= \|\mathbf{x} - F(\mathbf{y})\|_{\Omega}^2 = \text{tr}\{E_{xx} - E_{xy}F^T - FE_{yx} + FE_{yy}F^T\} \\ &= \|E_{xx}^{1/2}\|^2 - \|E_{xy}(E_{yy}^{1/2})^\dagger\|^2 + \|E_{xy}(E_{yy}^{1/2})^\dagger - FE_{yy}^{1/2}\|^2. \end{aligned} \tag{20}$$

Denote by $\mathbb{R}(m, n, k)$ the variety of all $m \times n$ matrices of rank at most k. For the sake of simplicity we also write $\mathbb{R}_k = \mathbb{R}(m, n, k)$.

In (20), only the last term depend on $F_1, \ldots, F_p$. Therefore, (9) and (20) imply

$$\min_{\mathcal{F}_1\in\mathcal{R}_{r_1},\ldots,\mathcal{F}_p\in\mathcal{R}_{r_p}} \left\|\mathbf{x} - \sum_{j=1}^{p} \mathcal{F}_j(\mathbf{y}_j)\right\|_{\Omega}^2 = \min_{F_1\in\mathbb{R}_{r_1},\ldots,F_p\in\mathbb{R}_{r_p}} \|E_{xy}(E_{yy}^{1/2})^\dagger - FE_{yy}^{1/2}\|^2. \tag{21}$$

Let us now denote $H = E_{xy}(E_{yy}^{1/2})^\dagger$ and represent matrix $E_{yy}^{1/2}$ in blocks, $E_{yy}^{1/2} = [G_1^T, \ldots, G_p^T]^T$ where $G_j \in \mathbb{R}^{n_j\times n}$, for $j = 1, \ldots, p$. Then in (21),

$$E_{xy}(E_{yy}^{1/2})^\dagger - FE_{yy}^{1/2} = H - \sum_{j=1}^{p} F_j G_j. \tag{22}$$

Therefore, (20) and (22) imply

$$\min_{\mathcal{F}_1\in\mathcal{R}_{r_1},\ldots,\mathcal{F}_p\in\mathcal{R}_{r_p}} \left\|\mathbf{x} - \sum_{j=1}^{p} \mathcal{F}_j(\mathbf{y}_j)\right\|_{\Omega}^2 = \min_{F_1\in\mathbb{R}_{r_1},\ldots,F_p\in\mathbb{R}_{r_1}} \left\|H - \sum_{j=1}^{p} F_j G_j\right\|^2. \tag{23}$$

On the basis of (23), problem (9) and the problem

$$\min_{F_1\in\mathbb{R}_{r_1},\ldots,F_p\in\mathbb{R}_{r_1}} \left\|H - \sum_{j=1}^{p} F_j G_j\right\|^2. \tag{24}$$

are equivalent. Therefore, below we consider problem (24). In (24), for $j = 1, \ldots, p$, we use the

representation

$$\left\| H - \sum_{j=1}^{p} F_j G_j \right\|^2 = \| S_j - F_j G_j \|^2 , \tag{25}$$

where $S_j = H - \sum_{\substack{i=1 \\ i \neq j}}^{p} F_i G_i$.

3) Greedy Method for Solution of Problem (24): A solution of problem (24) is based on the idea of the MBI method [1], [2] which is a greedy approach to solving optimization problems. The advantages of the MBI method have been mentioned in Section I-C. To begin with, let us denote

$$\mathbb{R}_{r_1,...,r_p} = \mathbb{R}_{r_1} \times ... \times \mathbb{R}_{r_p}, \quad \boldsymbol{F} = (F_1, ..., F_p) \in \mathbb{R}_{r_1,...,r_p} \quad \text{and} \quad f(\boldsymbol{F}) = \left\| H - \sum_{j=1}^{p} F_j G_j \right\|^2 .$$

The method we consider consists of the following steps.

1*st step.* Given $\boldsymbol{F}^{(0)} = (F_1^{(0)}, ..., F_p^{(0)}) \in \mathbb{R}_{r_1,...,r_p}$, compute, for $j = 1, \ldots, p$,

$$\widehat{F}_j^{(1)} = \left[S_j^{(0)} R_{G_j} \right]_{r_j} G_j^{\dagger}(I + K_j), \quad \text{where } S_j^{(0)} = H - \sum_{\substack{i=1 \\ i \neq j}}^{p} F_i^{(0)} G_i. \tag{26}$$

Note that $\widehat{F}_j^{(1)}$ is the solution of problem (16) represented by (17). A choice of $F_1^{(0)}, \ldots, F_p^{(0)}$, is considered in Section IV below.

2*nd step.* Denote

$$\bar{\boldsymbol{F}}_j^{(1)} = \left(F_1^{(0)}, \ldots, F_{j-1}^{(0)}, \widehat{F}_j^{(1)}, F_{j+1}^{(0)}, \ldots, F_p^{(0)} \right) , \tag{27}$$

select $\bar{\boldsymbol{F}}_k^{(1)}$ such that

$$f(\bar{\boldsymbol{F}}_k^{(1)}) = \min_{\bar{\boldsymbol{F}}_1^{(1)},...,\bar{\boldsymbol{F}}_p^{(1)}} \left\{ f(\bar{\boldsymbol{F}}_1^{(1)}), \ldots, f(\bar{\boldsymbol{F}}_k^{(1)}), \ldots, f(\bar{\boldsymbol{F}}_p^{(1)}) \right\} \tag{28}$$

and write

$$\boldsymbol{F}^{(1)} = \bar{\boldsymbol{F}}_k^{(1)}, \tag{29}$$

where we denote $\boldsymbol{F}^{(1)} = (F_1^{(1)}, ..., F_p^{(1)}) \in \mathbb{R}_{r_1,...,r_p}$.

Then we repeat procedure (26)-(29) with the replacement of $\boldsymbol{F}^{(0)}$ by $\boldsymbol{F}^{(1)}$ as follows: Given

$\boldsymbol{F}^{(1)} = (F_1^{(1)}, ..., F_p^{(1)})$, compute

$$\widehat{F}_j^{(2)} = \left[S_j^{(1)} R_{G_j}\right]_{r_j} G_j^{\dagger}(I + K_j), \quad \text{where } S_j^{(1)} = H - \sum_{\substack{i=1 \\ i \neq j}}^{p} F_i^{(1)} G_i,$$

select $\bar{\boldsymbol{F}}_k^{(2)}$ that satisfies (28) where superscript (1) is replaced with superscript (2), and set $\boldsymbol{F}^{(2)} = \bar{\boldsymbol{F}}_k^{(2)}$.

This process is continued up to the qth step when a given tolerance $\epsilon \geq 0$ is achieved in the sense

$$|f(\boldsymbol{F}^{(q+1)}) - f(\boldsymbol{F}^{(q)})| \leq \epsilon, \quad \text{for } q = 1, 2, \ldots. \tag{30}$$

It is summarized as follows.

Algorithm 1: Greedy solution of problem (24)

Initialization: $\boldsymbol{F}^{(0)}$, H, $S_1, ..., S_p$ and $\epsilon > 0$.

---

1. **for** $q = 0, 1, 2, ...$
2. **for** $j = 1, 2, ..., p$
3. $S_j^{(q)} = H - \sum_{\substack{i=1 \\ i \neq j}}^{p} F_i^{(q)} G_i$
4. $\widehat{F}_j^{(q+1)} = \left[S_j^{(q)} R_{G_j}\right]_{r_j} G_j^{\dagger}$
5. $\bar{\boldsymbol{F}}_j^{(q+1)} = \left(F_1^{(q)}, ..., F_{j-1}^{(q)}, \widehat{F}_j^{(q+1)}, F_{j+1}^{(q)}, ..., F_p^{(q)}\right)$
6. **end**
7. Choose $\boldsymbol{F}^{(q+1)} = \bar{\boldsymbol{F}}_k^{(q+1)}$ where $\boldsymbol{F}^{(q+1)} = (F_1^{(q+1)}, ..., F_p^{(q+1)})$ and $\bar{\boldsymbol{F}}_k^{(q+1)}$ is such that
$$f(\bar{\boldsymbol{F}}_k^{(q+1)}) = \min_{\bar{\boldsymbol{F}}_1^{(q+1)}, ..., \bar{\boldsymbol{F}}_p^{(q+1)}} \left\{ f(\bar{\boldsymbol{F}}_1^{(q+1)}), \ldots, f(\bar{\boldsymbol{F}}_k^{(q+1)}), \ldots, f(\bar{\boldsymbol{F}}_p^{(q+1)}) \right\}.$$
7. **If** $|f(\boldsymbol{F}^{(q+1)}) - f(\boldsymbol{F}^{(q)})| \leq \epsilon$
8. **Stop**
9. **end**
10. **end**

Algorithm 1 converges to a coordinate-wise minimum point of objective function $f(\boldsymbol{F}) =$

$\left\|H - \sum_{j=1}^{p} F_j G_j\right\|^2$ which is its local minimum. Section VIII-A provides more associated details.

Remark 1: For $p = 1$, the formulas for F_j in (17) and (26) coincide, and take the form

$$F_1 = [HR_{G_1}]_{r_1} G_1^{\dagger}(I + K_1) \tag{31}$$

where $G_1 = E_{y_1 y_1}^{1/2}$, $K_1 = M_1 (I - L_{G_1})$, $L_{G_1} = E_{y_1 y_1}^{1/2}(E_{y_1 y_1}^{\dagger})^{1/2}$ and $HR_{G_1} = E_{x y_1}(E_{y_1 y_1}^{\dagger})^{1/2}$. Thus, F_1 in (31) coincides with F_1 in (11) (since M_1 is arbitrary). In other words, the KLT represented by (11) is a particular case of the expressions in (17) and (26). For this reason, (17) and (26) can be regarded as extensions of the KLT to the case under consideration. Therefore, the transform represented by $\sum_{j=1}^{p} F_j(\mathbf{y}_j)$ where $F_1, \ldots, F_p$ solve (16) can be regarded the multi-compressor Karhunen-Loève-like transform (or the multi-compressor KLT). Further, Algorithm 1 represents the version of the MBI method that uses the multi-compressor KLT. Therefore, the WSN model in the form $\sum_{j=1}^{p} F_j^{(q+1)}(\mathbf{y}_j)$ where $F_1^{(q+1)}, \ldots, F_p^{(q+1)}$ are determined by Algorithm 1 can be interpreted as a transform as well. We call this transform the multi-compressor KLT-MBI.

Remark 2: In practice, an exact representation of matrices E_{xy} and E_{yy} might be unknown. Their estimates, $\widetilde{E}_{xy}$ and $\widetilde{E}_{yy}$, can be obtained by known methods [33], [34], [35], [36], [37], [38], [39], [40]. In this regard, we denote $\widetilde{H} = \widetilde{E}_{xy}(\widetilde{E}_{yy}^{1/2})^{\dagger}$ and $\widetilde{E}_{yy}^{1/2} = [\widetilde{G}_1^T, \ldots, \widetilde{G}_p^T]^T$ where $\widetilde{G}_j \in \mathbb{R}^{n_j \times n}$, for $j = 1, \ldots, p$, is a block of $\widetilde{E}_{yy}^{1/2}$. Then in (26), G_j, H and G_i should be replaced with $\widetilde{G}_j$, $\widetilde{H}$ and $\widetilde{G}_i$, respectively. In this case, steps (26)-(30) of the method and Algorithm 1 are the same as before but $\boldsymbol{F}^{(q+1)} = (F_1^{(q+1)}, ..., F_p^{(q+1)})$ should be denoted by $\widetilde{\boldsymbol{F}}^{(q+1)} = (\widetilde{F}_1^{(q+1)}, ..., \widetilde{F}_p^{(q+1)})$.

B. Models of Sensors and Fusion Center

The models of sensors $Q_1, \ldots, Q_p$ and the fusion center $P = [P_1, \ldots, P_p]$ of the WSN in Fig. 1 follow from the formula $F_j = P_j Q_j$ introduced in (8). By the proposed method, F_j is represented by $F_j^{(q+1)}$ given by Algorithm 1 above. Therefore, the mathematical model of the

WSN is given by

$$\sum_{j=1}^{p} F_j^{(q+1)}(\mathbf{y}_j) = \sum_{j=1}^{p} P_j^{(q+1)} Q_j^{(q+1)}(\mathbf{y}_j) = P^{(q+1)} \left(\begin{array}{c} Q_1^{(q+1)}(\mathbf{y}_1) \\ \vdots \\ Q_p^{(q+1)}(\mathbf{y}_p) \end{array} \right). \tag{32}$$

Here, $F_j^{(q+1)} = P_j^{(q+1)} Q_j^{(q+1)}$ and $P^{(q+1)} = [P_1^{(q+1)}, \ldots, P_p^{(q+1)}]$. Then $Q_j^{(q+1)} \in \mathbb{R}^{r_j \times n_j}$ and $P_j^{(q+1)} \in \mathbb{R}^{m \times r_j}$ follow from the SVD decomposition of matrix $F_j^{(q+1)}$ taken with r_j first nonzero singular values. This procedure has been described in Section II-D for the case of only one sensor, i.e. for $p = 1$.

For the case when instead of matrices E_{xy} and E_{yy} their estimates $\widetilde{E}_{xy}$ and $\widetilde{E}_{yy}$ are used, the models are constructed by the similar procedure with the replacement of $F_j^{(q+1)}$ by $\widetilde{F}_j^{(q+1)}$ (see Remark 2 above).

Note that the proposed WSN model also provides de-noising of observations $\mathbf{y}_1, \ldots, \mathbf{y}_p$.

IV. Determination of Initial Iterations

To start Algorithm 1, the values of initial iterations $F_j^{(0)}$ and $\widetilde{F}_j^{(0)}$ should be defined. It is done as follows. Let us denote $\mathbf{x} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_p^T]^T$ where $\mathbf{x}_i \in L^2(\Omega, \mathbb{R}^{m_i})$, $i = 1, \ldots, p$ and $m_1 + \ldots + m_p = m$. Suppose that matrix $P \in \mathbb{R}^{m \times r}$ is given by $P = \mathrm{diag}(P_{11}, \ldots, P_{pp})$ where $P_{jj} \in \mathbb{R}^{m_j \times r_j}$, for $j = 1, \ldots, p$. Then

$$\left\| \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_p \end{bmatrix} - \mathrm{diag}(\mathcal{P}_{11}, \ldots, \mathcal{P}_{pp}) \begin{bmatrix} \mathcal{Q}_1(\mathbf{y}_1) \\ \vdots \\ \mathcal{Q}_p(\mathbf{y}_p) \end{bmatrix} \right\|_\Omega^2$$

$$= \left\| \begin{bmatrix} \mathbf{x}_1 - \mathcal{F}_1(\mathbf{y}_1) \\ \vdots \\ \mathbf{x}_p - \mathcal{F}_p(\mathbf{y}_p) \end{bmatrix} \right\|_\Omega^2 = \sum_{j=1}^{p} \|\mathbf{x}_j - \mathcal{F}_j(\mathbf{y}_j)\|_\Omega^2 \tag{33}$$

and

$$\min_{\mathcal{F}_1 \in \mathcal{R}_{r_1}, \ldots, \mathcal{F}_p \in \mathcal{R}_{r_p}} \sum_{j=1}^{p} \|\mathbf{x}_j - F_j(\mathbf{y}_j)\|_\Omega^2 = \sum_{j=1}^{p} \min_{F_j \in \mathcal{R}_{r_j}} \|\mathbf{x}_j - \mathcal{F}_j(\mathbf{y}_j)\|_\Omega^2. \tag{34}$$

As a result, in this case, problem (9) is reduced to the problem of finding $\mathcal{F}_j$ that solves

$$\min_{\mathcal{F}_j \in \mathcal{R}_{r_j}} \left\| \mathbf{x}_j - \mathcal{F}_j(\mathbf{y}_j) \right\|_\Omega^2, \tag{35}$$

for $j = 1, \ldots, p$. Its solution is given by (11) in Section II-D above, i.e. by

$$\widehat{F}_j = \left[E_{x_j y_j} (E^\dagger_{y_j y_j})^{1/2} \right]_{r_j} (E^\dagger_{y_j y_j})^{1/2} + M_j [I - E^{1/2}_{y_j y_j} (E^\dagger_{y_j y_j})^{1/2}] \tag{36}$$

where M_j is an arbitrary matrix. Then the initial iterations for Algorithm 1 are defined by

$$F_j^{(0)} = \widehat{F}_j, \quad \text{for } j = 1, \ldots, p. \tag{37}$$

Similarly, when instead of matrices E_{xy} and E_{yy} their estimates $\widetilde{E}_{xy} = \{\widetilde{E}_{x_i y_j}\}_{i,j=1}^p$ and $\widetilde{E}_{yy} = \{\widetilde{E}_{y_i y_j}\}_{i,j=1}^p$ are used, the initial iterations are defined by

$$\widetilde{F}_j^{(0)} = \left[\widetilde{E}_{x_j y_j} (\widetilde{E}^\dagger_{y_j y_j})^{1/2} \right]_{r_j} (\widetilde{E}^\dagger_{y_j y_j})^{1/2} + M_j [I - \widetilde{E}^{1/2}_{y_j y_j} (\widetilde{E}^\dagger_{y_j y_j})^{1/2}], \tag{38}$$

where $\widetilde{E}_{x_j y_j}$ and $\widetilde{E}_{y_j y_j}$ are blocks of $\widetilde{E}_{xy}$ and $\widetilde{E}_{yy}$, respectively.

V. Error Analysis: A Posteriori Associated Errors

For $F^{(q+1)} = [F_1^{(q+1)}, \ldots, F_p^{(q+1)}]$ determined by Algorithm 1, the error associated with the proposed WSN model is represented as

$$\|\mathbf{x} - [F_1^{(q+1)}, \ldots, F_p^{(q+1)}](\mathbf{y})\|_\Omega^2 = \|E_{xx}^{1/2}\|^2 - \|E_{xy}(E_{yy}^{1/2})^\dagger\|^2 + \|E_{xy}(E_{yy}^{1/2})^\dagger - F^{(q+1)} E_{yy}^{1/2}\|^2,$$

where $F^{(q+1)} = [F_1^{(q+1)}, \ldots, F_p^{(q+1)}]$.

For $\widetilde{F}_j^{(q+1)}$ described in Remark 2, the associated error is given by the similar expression:

$$\|\mathbf{x} - [\widetilde{F}_1^{(q+1)}, \ldots, \widetilde{F}_p^{(q+1)}](\mathbf{y})\|_\Omega^2 = \|\widetilde{E}_{xx}^{1/2}\|^2 - \|\widetilde{E}_{xy}(\widetilde{E}_{yy}^{1/2})^\dagger\|^2 + \|\widetilde{E}_{xy}(\widetilde{E}_{yy}^{1/2})^\dagger - \widetilde{F}^{(q+1)} \widetilde{E}_{yy}^{1/2}\|^2, \tag{39}$$

where $\widetilde{F}^{(q+1)} = [\widetilde{F}_1^{(q+1)}, \ldots, \widetilde{F}_p^{(q+1)}]$. The above formula (39) is used in our simulations represented in the following section.

VI. Simulations

Here, we wish to illustrate the advantages of the proposed methodology with numerical examples carried out under the assumption that either covariance matrices E_{xy}, E_{yy} or their

estimates are known. The assumption that only the covariance matrices are known is similar to that used in [6], [7], [8], [10], [11], [17], [21], [22], [23], [24], [26], [27]. In particular, the estimates can be obtained from samples of training signals. In many situations, the number of samples, s, is often smaller than the dimensions of the signals $\mathbf{x}$ and $\mathbf{y}$, which are m and n, respectively [38]. At the same time, it is known that as $s \to \infty$, the ergodic theorem asserts that the estimates converge to the true matrix values [39], [40]. In particular, for large s, the estimates of the covariance matrix have been considered in [34], [35]. It is interesting to compare our simulation results for the cases when s is 'relatively' small and 'relatively' large. In the examples that follow, both case are considered.

A comparison with known methods [11], [17], [21], [22], [23], [24] is a s follows. The method [24] represents a generalization of methods [21], [22], [23] and therefore, we provide a numerical comparison with method [24] which includes, in fact, a comparison with methods [21], [22], [23] as well. Further, covariance matrices used in the simulations associated with Figs. 3 (a), (c) and Figs. 4 (b), (c), (d) are singular, and therefore, method [17] is not applicable (in this regard, see also Section II-C). Therefore, in Figs. 3 (a), (c) and Figs. 4 (b), (c), (d), results related to algorithm in [17] are not given. By the same reason, the method presented in [11] is not applicable as well. Moreover, the method in [11] is restricted to the case when the covariance matrix formed by the noise vector is block diagonal which is not the case here.

In the examples below, different types of noisy observed signals and different compression ratios are considered. In all examples, our method provides the better associated accuracy than that for the methods in [17], [24] (and methods in [21], [22], [23] as well, because they follow from [24]).

Example 1: We start with an example similar to that considered in [24] assuming that a WSN has two sensors and the observations $\mathbf{y}_1$ and $\mathbf{y}_2$ are represented by

$$\mathbf{y}_1 = \mathbf{x} + \xi_1, \quad \text{and} \quad \mathbf{y}_2 = \mathbf{x} + \xi_2, \tag{40}$$

where $\mathbf{x} \in L^2(\Omega, \mathbb{R}^3)$, $\xi_1 \in L^2(\Omega, \mathbb{R}^3)$ and $\xi_2 \in L^2(\Omega, \mathbb{R}^3)$ are Gaussian *independent* random vectors with the zero mean. Let $E_{xx} = \begin{bmatrix} 0.585 & 0.270 & 0.390 \\ 0.270 & 0.405 & 0.180 \\ 0.390 & 0.180 & 0.260 \end{bmatrix}$ and $E_{\xi_j,\xi_j} = \sigma_j^2 I_3$, for $j = 1, 2$, where $\sigma_1 = 0.2$ and $\sigma_2 = 0.4$, and I_3 is the 3×3 identity matrix. Then $E_{xy} = [E_{xx}\ E_{xx}]$ and

$E_{yy} = \begin{bmatrix} E_{xx}+\sigma_1^2 I_3 & E_{xx} \\ E_{xx} & E_{xx}+\sigma_2^2 I_3 \end{bmatrix}$. For $r_1 = r_2 = 1$, Algorithm 1 requires three iterations to achieve tolerance $\epsilon = 0.146$. The achievable tolerance of methods [17], [24], for $r_1 = r_2 = 1$, is 18% worse, $\epsilon = 0.173$, and it is not improved after the initial iteration proposed in [24].

Example 2: Let us consider the case of a WSN with two sensors again where, as before,

$$\mathbf{y}_1 = \mathbf{x} + \xi_1, \quad \text{and} \quad \mathbf{y}_2 = \mathbf{x} + \xi_2, \tag{41}$$

where $\mathbf{x} \in L^2(\Omega, \mathbb{R}^m)$, $\xi_1 \in L^2(\Omega, \mathbb{R}^m)$ and $\xi_2 \in L^2(\Omega, \mathbb{R}^m)$, and $\mathbf{y}_1$ and $\mathbf{y}_2$ are noisy versions of the source. Unlike Example 1 we now assume that covariance matrices E_{xy} and E_{yy} are unknown. Therefore, their estimates $\widetilde{E}_{xy}$ and $\widetilde{E}_{yy} = \left\{\widetilde{E}_{y_i y_j}\right\}$, $i, j = 1, 2$, should be used. To this end, estimates $\widetilde{E}_{xy}$ and $\widetilde{E}_{yy}$ have been determined from the samples of training signals as follows:

$$\widetilde{E}_{xy} = \frac{1}{s}[XY_1^T, XY_2^T] \quad \text{and} \quad \widetilde{E}_{y_i y_j} = \frac{1}{s} Y_i Y_j^T \quad \text{for } i, j = 1, 2. \tag{42}$$

Here, $X \in \mathbb{R}^{m \times s}$ has uniformly distributed random entries and, for $i = 1, 2$,

$$Y_i = X + \sigma_i \varUpsilon_i, \tag{43}$$

where $\sigma_i \in \mathbb{R}$ and $\varUpsilon_i \in \mathbb{R}^{m \times s}$ has random entries, chosen from a normal distribution with mean zero and variance one. Diagrams of typical errors associated with the proposed Algorithm 1 and known methods [17], [24] are given in Figs. 3 (a), (b).

Note that in Figs. 3 (a), (b), the obtained results are illustrated for different compression ratios $c_j = r_j / n_j$ where $j = 1, \ldots, p$. The compression ratios $c_j = 1/5$, for $j = 1, 2, 3$, used to obtain the results represented in Fig. 3 (b) are smaller than those in Fig. 3 (a), $c_1 = 3/5$ and $c_2 = 7/10$. This is a reason for the error magnitudes represented in Fig. 3 (a) being smaller than those in Fig. 3 (b). This observation also holds for other examples that follow. Further, in Fig. 3 (b), due to large sample size, $s = 10000$, the estimate of matrix $E_{y_i y_j}$ is very close to its true value which is the identity. By this reason, iterations of our method are similar to each other and the associated errors are similar for almost all iterations. The same effect holds for methods [17], [24].

Example 3: Here, we consider the case when observations $\mathbf{y}_1$ and $\mathbf{y}_2$ are very noisy, i.e.

reference signal $\mathbf{x}$ is significantly suppressed. To this end, we do not assume that $Y_i \in \mathbb{R}^{m\times s}$ is represented in the form (43) but it has random entries, chosen from a normal distribution with mean zero and variance one. We also use $\widetilde{E}_{xy}$ and $\widetilde{E}_{yy} = \left\{\widetilde{E}_{y_iy_j}\right\}$, $i,j = 1,2$ in the form (42) as before where $X \in \mathbb{R}^{m\times s}$ is as in the above Example 2. For $m = n_1 = N_2 = 2$ and $s = 4$, examples of those matrices are $X = \begin{bmatrix} 0.086 & 0.439 & 0.857 & 0.904 \\ 0.074 & 0.574 & 0.386 & 0.429 \end{bmatrix}$, $Y_1 = \begin{bmatrix} 0.284 & -0.942 & 0.067 & 0.222 \\ -2.206 & 0.514 & -1.293 & -0.686 \end{bmatrix}$ and $Y_2 = \begin{bmatrix} 0.4660 & -0.1260 & 0.3870 & 0.3290 \\ 0.6880 & -0.4690 & -0.9420 & -0.5630 \end{bmatrix}$.

For the case of two sensors (i.e. for $p = 2$) and for the above samples X, Y_1 and Y_2, the errors associated with the proposed method and known method [24] are given in Fig. 3 (c).

For larger magnitudes of m, n_1, n_2, n_3, s and r_j, for $j = 1, 2, 3$, the errors associated with the proposed method and known methods are represented, for the case of two and three sensors (i.e. for $p = 2$ and $p = 3$, respectively), in Fig. 3 (d) and Fig. 4 (a).

Example 4: In this example, we consider the case when observations are corrupted by noise in the way which is different from those in Examples 2 and 3. Namely, we assume that, for $j = 1, \ldots, p$,

$$\mathbf{y}_j = \mathcal{A}_j\mathbf{x} + \xi_j \tag{44}$$

where $\mathcal{A}_j : L^2(\Omega, \mathbb{R}^m) \to L^2(\Omega, \mathbb{R}^m)$ is a linear operator defined by matrix $A_j \in \mathbb{R}^{m\times m}$ with uniformly distributed random entries, and ξ_j is a random noise. Samples of $\mathbf{x}$ and ξ_j are simulated as matrices $X \in \mathbb{R}^{m\times s}$ and $\sigma_j \Upsilon_j \in \mathbb{R}^{m\times s}$, respectively, where $\sigma_j \in \mathbb{R}$, such that X has uniformly distributed random entries and Υ_j has random entries, chosen from a normal distribution with mean zero and variance one.

The errors associated with the proposed method and the known method, for the case of two and three sensors (i.e. for $p = 2$ and $p = 3$, respectively), and different choices of m, n_j, s and r_j, for $j = 1, 2$ and $j = 1, 2, 3$, are represented in Figs. 4 (b), (c) and (d).

Note that method [24] is not numerically stable in these simulations. We believe this is because of the reason mentioned in Section I-C.

Example 5: In the above examples, we used estimates of training signals, not training signals themselves. Here, we wish to illustrate the obtained theoretical results in a different way, by a comparison of a training reference signal with its estimates obtained by our method and known

methods. To this end, we simulate the training reference signal $\mathbf{x}$ by its realizations, i.e. by a matrix $\boldsymbol{X} \in \mathbb{R}^{m\times k}$ where each column represents a realization of the signal. A sample $X \in \mathbb{R}^{m\times s}$ with $s < k$ is formed from $\boldsymbol{X}$ by choosing the even columns. To represent the obtained results in a visible way, signal $\boldsymbol{X} \in \mathbb{R}^{m\times k}$ is chosen as the known image Lena given by the 128×128 matrix – see Fig. 5 (a), i.e. with $m, k = 128$. Then $X \in \mathbb{R}^{128\times 64}$.

Further, we consider the WSN with two sensors, i.e. with $p = 2$, where the observed signal $\boldsymbol{Y}_j$, for $j = 1, 2$, is simulated as follows:

$$\boldsymbol{Y}_j = A_j {*} \boldsymbol{X} + \sigma_j \varUpsilon_j$$

where $A_j \in \mathbb{R}^{128\times 128}$ has uniformly distributed random entries, $\varUpsilon_j \in \mathbb{R}^{128\times 128}$ has random entries, chosen from a normal distribution with mean zero and variance one, $A_j {*} X$ represents the Hadamard matrix product, and $\sigma_1 = 0.2$ and $\sigma_2 = 0.1$. Estimates $\widetilde{E}_{xy}$ and $\widetilde{E}_{y_i y_j}$ are used in the form (42) where sample $Y_j \in \mathbb{R}^{128\times 64}$ is formed from $\boldsymbol{Y}_j$ by choosing the even columns.

For $r_1 = r_2 = 64$, the simulation results are represented in Figs. 5 and 6. Our method and known methods in [24] and [17]) have been applied to the above signals with 50 iterations each. The associated errors are evaluated in the form $|\boldsymbol{X} - \widehat{\boldsymbol{X}}|$ where $\widehat{\boldsymbol{X}}$ is the reconstruction of $\boldsymbol{X}$ by the method we use (i.e. by our method or methods in [17] and [24]).

Similar to the other examples, Figs. 5 and 6 demonstrate a more accurate signal reconstruction associated with the proposed method than that associated with known methods.

VII. Conclusion

We have addressed the problem of estimating an unknown random vector source when the vector cannot be observed centrally. In this scenario, typical of wireless sensor networks (WSNs), distributed sensors are aimed to filter and compress noisy observed vector, and then the compressed signals are transmitted to the fusion center that decompress the signals in such a way that the original vector is estimated within a prescribed accuracy. The key problem is to find models of the sensors and the fusion center in the best possible way.

We proposed and justified the method for the determination of the models based on a combination of the solution [5] of the rank constrained least squares minimizing problem (represented by (9) in Section II-B) and the maximum block improvement (MBI) method [1], [2]. The proposed method is based on the following steps. First, we have shown how the original problem can be

reduced to the form (21) (Section III-A2) that allowed us to use the approaches developed in [1], [2], [5]. As a result, under the assumption that the associated covariance matrices or their estimates are known (from testing experiments, for example), the procedure for determining models of the sensors and the fusion center is given by Algorithm 1 (Section III-A3).

The obtained optimal WSN model represents an extension of the Karhunen-Loève transform (KLT) and has been called the multi-compressor KLT-MBI. The known KLT follows from the multi-compressor KLT-MBI as a particular case. The models of the sensors and the fusion center have been determined in terms of the pseudo-inverse matrices. Therefore, the proposed models are always well determined and numerically stable. In other words, the proposed WSN models provide compression, de-noising and reconstruction of distributed signals for the cases when known methods either are not applicable or produce larger associated errors. As a result, this approach mitigates to some extent the difficulties associated with the existing techniques. Since a 'good' choice of the initial iteration gives reduced errors, the special method for the determination of the initial iterations has been considered.

The error analysis of the proposed method has been provided.

Finally, the advantages of the proposed method have been illustrated with numerical experiments carried out on the basis of simulations with estimates of the covariance matrices. It has been shown, in particular, that the errors associated with the proposed technique are smaller than those associated with the existing methods. This is because of the special features of our method described above.

VIII. Appendix

A. Convergence

Convergence of the method presented in Section III-A3 can be shown on the basis of the results presented in [1], [2] as follows.

We call $\boldsymbol{F} = (F_1, ..., F_p) \in \mathbb{R}_{r_1,...,r_p}$ a point in the space $\mathbb{R}_{r_1,...,r_p}$. For every point $\boldsymbol{F} \in \mathbb{R}_{r_1,...,r_p}$, define a set

$$\mathbb{R}^{\boldsymbol{F}}_{r_j} = \{(F_1, \dots, F_{j-1})\} \times \mathbb{R}_{r_j} \times \{(F_{j+1}, \dots, F_p)\}, \quad \text{for } j = 1, \dots, p.$$

A coordinate-wise minimum point of the procedure represented by Algorithm 1 is denoted by

$\boldsymbol{F}^* = \left(F_1^*, ..., F_p^*\right)$ where[5]

$$F_j^* \in \left\{ \arg \min_{F_j \in \mathbb{R}_{r_j}} \; f(F_1^*, ..., F_{j-1}^*, F_j, F_{j+1}^*, ..., F_p^*) \right\}. \tag{45}$$

This point is a local minimum of objective function in (24), $f(\boldsymbol{F}) = \left\| H - \sum_{j=1}^{p} F_j G_j \right\|^2$.[6] Note that $\boldsymbol{F}^{(q+1)}$ in Algorithm 1 and $\boldsymbol{F}^*$ defined by (45) are, of course, different.

For $\boldsymbol{F}^{(q)}$ defined by Algorithm 1, denote

$$\check{\boldsymbol{F}} = \lim_{q \to \infty} \boldsymbol{F}^{(q)}. \tag{46}$$

Note that because of (5), the sequence $\{\boldsymbol{F}^{(q)}\}$ is bounded.

Theorem 2: Point $\check{\boldsymbol{F}}$ defined by (46) is the coordinate-wise minimum of Algorithm 1.

Proof: For each fixed $\boldsymbol{F} = (F_1, ..., F_p)$, a so-called best response matrix to matrix F_j is denoted by $\varPhi_j^{\boldsymbol{F}}$, where

$$\varPhi_j^{\boldsymbol{F}} \in \left\{ \arg \min_{F_j \in \mathbb{R}_{r_j}} \; f(F_1, ...F_{j-1}, F_j, F_{j+1}, ..., F_p) \right\}.$$

Let $\{\boldsymbol{F}^{(q)}\}$ be a sequence generated by Algorithm 1, where $\boldsymbol{F}^{(q)} = (F_1^{(q)}, ..., F_p^{(q)})$. Since each $\mathbb{R}_{r_j}$ is closed [41, p. 304], there is a subsequence $\{\boldsymbol{F}^{(q_s)}\}$ such that $(F_1^{(q_s)}, ..., F_p^{(q_s)}) \to (F_1^*, ..., F_p^*) = \boldsymbol{F}^*$ as $s \to \infty$. Then, for any $j = 1, ..., p$, we have

$$\begin{aligned} f(F_1^{(q_s)}, ..., F_{j-1}^{(q_s)}, \varPhi_j^{\boldsymbol{F}^*}, F_{j+1}^{(q_s)}, ..., F_p^{(q_s)}) \;&\geq\; f(F_1^{(q_s)}, ..., F_{j-1}^{(q_s)}, \varPhi_j^{\boldsymbol{F}^{(q_s)}}, F_{j+1}^{(q_s)}, ..., F_p^{(q_s)}) \\ &\geq\; f(F_1^{(q_s+1)}, ..., F_{j-1}^{(q_s+1)}, F_j^{(q_s+1)}, F_{j+1}^{(q_s+1)}, ..., F_p^{(q_s+1)}) \\ &\geq\; f(F_1^{(q_{s+1})}, ..., F_{j-1}^{(q_{s+1})}, F_j^{(q_{s+1})}, F_{j+1}^{(q_{s+1})}, ..., F_p^{(q_{s+1})}) \end{aligned}$$

By continuity, when $s \to \infty$,

$$f(F_1^*, ..., F_{j-1}^*, \varPhi_j^{\boldsymbol{F}^*}, F_{j+1}^*, ..., F_p^*) \geq f(F_1^*, ..., F_{j-1}^*, F_j^*, F_{j+1}^*, ..., F_p^*),$$

which implies that above should hold as an equality, since the inequality is true by the definition

[5]The RHS in (45) is a set since the solution of problem $\min_{F_j \in \mathbb{R}_{r_j}} \; f(F_1^*, ..., F_{j-1}^*, F_j, F_{j+1}^*, ..., F_p^*)$ is not unique.

[6]There could be other local minimums defined differently from that in (45).

of the best response matrix $\Phi_j^{\boldsymbol{F}^*}$. Thus, F_j^* is such as in (45), i.e. F_j^* is a solution of the problem

$$\min_{F_j \in \mathcal{R}_{r_j}} f(F_1^*, ..., F_{j-1}^*, F_j, F_{j+1}^*, ..., F_p^*), \ \forall j = 1, ..., p.$$

■

Remark 3: Theorem 2 still holds if the objective function is defined as $f(\widetilde{\boldsymbol{F}}) = \left\| \widetilde{H} - \sum_{j=1}^{p} \widetilde{F}_j \widetilde{G}_j \right\|^2$ where $\widetilde{\boldsymbol{F}} = (\widetilde{F}_1, \ldots, \widetilde{F}_p)$, and $\widetilde{H}$ and $\widetilde{G}_j$ are defined by Remark 2. In this case, the coordinate-wise minimum point is defined similar to that in (45) where symbols $\boldsymbol{F}^*$, F_j^* and F_j should be replaced with $\widetilde{\boldsymbol{F}}^*$, $\widetilde{F}_j^*$ and $\widetilde{F}_j$, respectively. More precisely, if $\widetilde{\boldsymbol{F}}^{(q)}$ denotes the qth iteration of Algorithm 1 as described in Remark 2, then the following is true.

Corollary 1: Point $\widehat{\boldsymbol{F}}$ defined by

$$\widehat{\boldsymbol{F}} = \lim_{q \to \infty} \widetilde{\boldsymbol{F}}^{(q)}$$

is the coordinate-wise minimum of Algorithm 1 in the case when covariance matrices E_{xy} and E_{yy} are replaced with their estimates $\widetilde{E}_{xy}$ and $\widetilde{E}_{yy}$, respectively.

REFERENCES

[1] B. Chen, S. He, Z. Li, S. Zhang, Maximum block improvement and polynomial optimization, SIAM Journal on Optimization 22 (1) (2012) 87–107.

[2] Z. Li, A. Uschmajew, S. Zhang, On convergence of the maximum block improvement method, SIAM Journal on Optimization 25 (1) (2015) 210–233.

[3] A. Torokhti, P. Howlett, Computational Methods for Modelling of Nonlinear Systems, Elsevier, 2007.

[4] S. Friedland, A. Torokhti, Generalized rank-constrained matrix approximations, SIAM Journal on Matrix Analysis and Applications 29 (2) (2007) 656–659.

[5] A. Torokhti, S. Friedland, Towards theory of generic Principal Component Analysis, Journal of Multivariate Analysis 100 (4) (2009) 661 – 669.

[6] P. L. Dragotti, M. Gastpar, Distributed Source Coding: Theory, Algorithms and Applications, Academic Press, 2009.

[7] J. Fang, H. Li, Optimal/near-optimal dimensionality reduction for distributed estimation in homogeneous and certain inhomogeneous scenarios, IEEE Transactions on Signal Processing 58 (8) (2010) 4339–4353.

[8] A. Amar, A. Leshem, M. Gastpar, Recursive implementation of the distributed Karhunen-Loeve transform, IEEE Transactions on Signal Processing 58 (10) (2010) 5320–5330.

[9] A. Bertrand, M. Moonen, Distributed adaptive node-specific signal estimation in fully connected sensor networkspart i: Sequential node updating, IEEE Transactions on Signal Processing 58 (10) (2010) 5277 – 5291.

[10] M. Lara, B. Mulgrew, Performance of the distributed KLT and its approximate implementation, in: 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO), 2012, pp. 724–728.

[11] H. Ma, Y.-H. Yang, Y. Chen, K. Liu, Q. Wang, Distributed state estimation with dimension reduction preprocessing, Signal Processing, IEEE Transactions on 62 (12) (2014) 3098–3110.

[12] D. E. Marelli, M. Fu, Distributed weighted least-squares estimation with fast convergence for large-scale systems, Automatica 51 (0) (2015) 27 – 39.

[13] L. L. Scharf, The SVD and reduced rank signal processing, Signal Processing 25 (2) (1991) 113 – 133.

[14] Y. Hua, W. Liu, Generalized Karhunen-Loeve transform, IEEE Signal Processing Letters 5 (6) (1998) 141–142.

[15] Y. Hua, M. Nikpour, P. Stoica, Optimal reduced-rank estimation and filtering, IEEE Transactions on Signal Processing 49 (3) (2001) 457–469.

[16] A. Torokhti, S. Miklavcic, Data compression under constraints of causality and variable finite memory, Signal Processing 90 (10) (2010) 2822 – 2834.

[17] I. D. Schizas, G. B. Giannakis, Z.-Q. Luo, Distributed estimation using reduced-dimensionality sensor observations, IEEE Transactions on Signal Processing 55 (8) (2007) 4284–4299.

[18] A. Torokhti, P. Howlett, Optimal fixed rank transform of the second degree, IEEE Trans. CAS. Part II, Analog and Digital Signal Processing 48 (3) (2001) 309 – 315.

[19] D. Slepian, J. Wolf, Noiseless coding of correlated information sources, IEEE Transactions on Information Theory 19 (4) (1973) 471–480.

[20] A. Wyner, J. Ziv, The rate-distortion function for source coding with side information at the decoder, IEEE Transactions on Information Theory 22 (1) (1976) 1–10.

[21] E. Song, Y. Zhu, J. Zhou, Sensors optimal dimensionality compression matrix in estimation fusion, Automatica 41 (12) (2005) 2131 – 2139.

[22] Y. Zhu, E. Song, J. Zhou, Z. You, Optimal dimensionality reduction of sensor data in multisensor estimation fusion, IEEE Transactions on Signal Processing 53 (5) (2005) 1631–1639.

[23] M. Gastpar, P. Dragotti, M. Vetterli, The distributed Karhunen-Loève transform, IEEE Transactions on Information Theory 52 (12) (2006) 5177–5196.

[24] O. Roy, M. Vetterli, Dimensionality reduction for distributed estimation in the infinite dimensional regime, IEEE Transactions on Information Theory 54 (4) (2008) 1655–1669.

[25] V. Goyal, Theoretical foundations of transform coding, IEEE Signal Processing Magazine 18 (5) (2001) 9–21.

[26] I. D. Schizas, A. Ribeiro, G. B. Giannakis, Dimensionality reduction, compression and quantization for distributed estimation with wireless sensor networks, in: Wireless Communications, Vol. 143 of The IMA Volumes in Mathematics and its Applications, Springer New York, 2007, pp. 259–296.

[27] J. A. Saghri, S. Schroeder, A. G. Tescher, Adaptive two-stage Karhunen-Loeve-transform scheme for spectral decorrelation in hyperspectral bandwidth compression, Optical Engineering 49 (5) (2010) 057001–057001–7.

[28] V. Ejov, A. Torokhti, How to transform matrices U_1,...,U_p to matrices V_1,...,V_p so that $V_i V_j = \mathbb{O}$ if $i \neq j$?, Numerical Algebra, Control and Optimization 2 (2) (2012) 293–299.

[29] D. R. Brillinger, Time Series: Data Analysis and Theory, Holden Day, San Francisco, 2001.

[30] A. Torokhti, S. Friedland, Towards theory of generic principal component analysis, Journal of Multivariate Analysis 100 (4) (2009) 661 – 669.

[31] P. Tseng, Convergence of a block coordinate descent method for nondifferentiable minimization, Journal of Optimization Theory and Applications 109 (3) (2001) 475–494.

[32] D. Bertsekas, Nonlinear Programming, Athena Scientific, 1995.

[33] L. Perlovsky, T. Marzetta, Estimating a covariance matrix from incomplete realizations of a random vector, IEEE Transactions on Signal Processing 40 (8) (1992) 2097–2100.

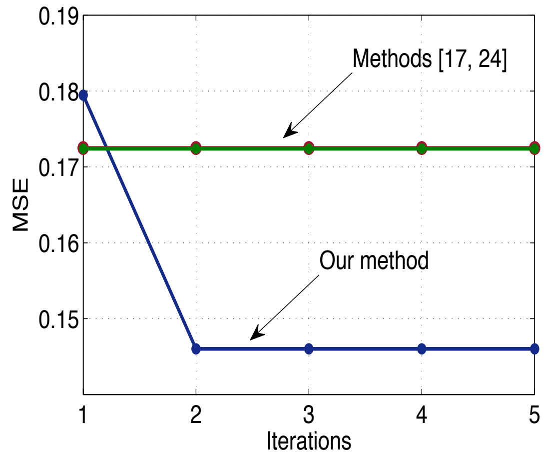


Fig. 2. Example 1: Diagrams of MSE's associated with the proposed method and the known methods versus number of iterations.

[34] O. Ledoit, M. Wolf, A well-conditioned estimator for large-dimensional covariance matrices, Journal of Multivariate Analysis 88 (2) (2004) 365 – 411.

[35] O. Ledoit, M. Wolf, Nonlinear shrinkage estimation of large-dimensional covariance matrices, Ann. Statist. 40 (2) (2012) 1024–1060.

[36] R. Adamczak, A. E. Litvak, A. Pajor, N. Tomczak-Jaegermann, Quantitative estimates of the convergence of the empirical covariance matrix in log-concave ensembles, Journal of the American Mathematical Society (23) (2009) 535–561.

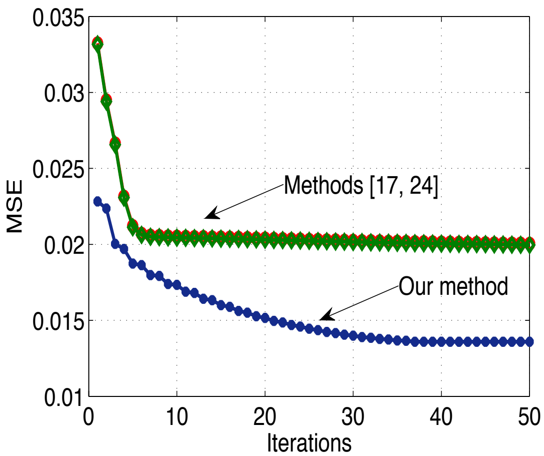
[37] R. Vershynin, How close is the sample covariance matrix to the actual covariance matrix?, Journal of Theoretical Probability 25 (3) (2012) 655–686.

[38] S.-J. K. Joong-Ho Won, Johan Lim, B. Rajaratnam, Condition-number-regularized covariance estimation, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 75 (3) (2013) 427–450.

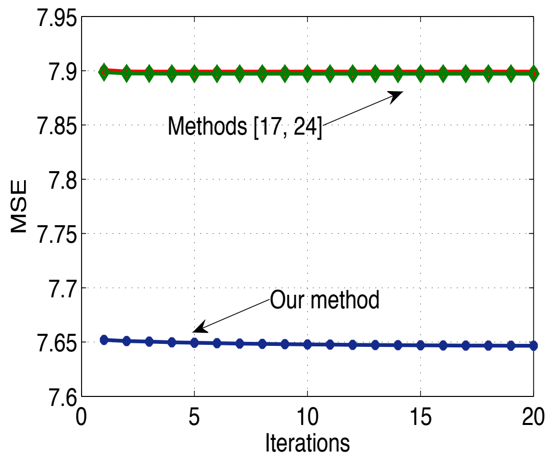
[39] B. W. Schmeiser, M. H. Chen, On hit-and-run Monte Carlo sampling for evaluating multidimensional integrals, Technical Report 91-39, Dept. Statistics, Purdue Univ.

[40] R. Yang, J. O. Berger, Estimation of a covariance matrix using the reference prior, The Anna1s of Statistics 22 (3) (1994) 1195–1211.

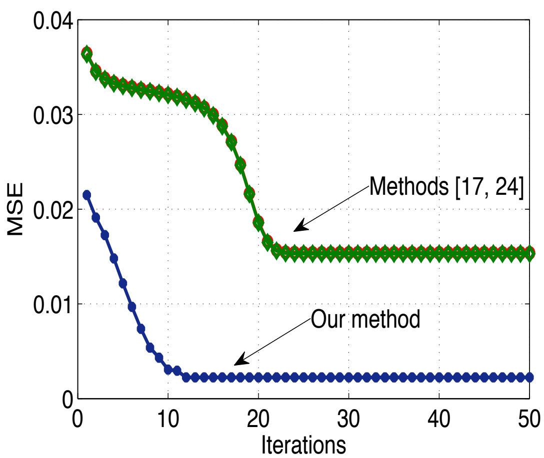
[41] L. Tu, An Introduction to Manifolds, Universitext, Springer, 2007.



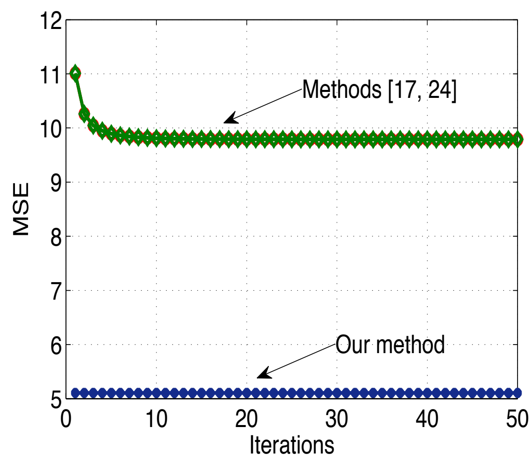
(a) Example 2: $p = 2$, $m = n_j = 10$, $s = 20$, $r_j = 5 + j$, $\sigma_j = 0.2 - 0.1j$, for $j = 1, 2,$.



(b) Example 2: $p = 3$, $m = n_j = 100$, $s = 10000$, $r_j = 20$, $\sigma_j = 1$, for $j = 1, 2, 3$.

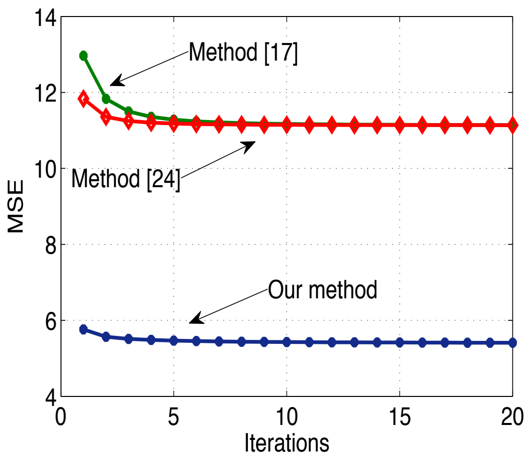


(c) Example 3: $p = 2$, $m = n_j = 2$, $s = 4$, $r_j = 1$, $\sigma_j = 1$, for $j = 1, 2$.

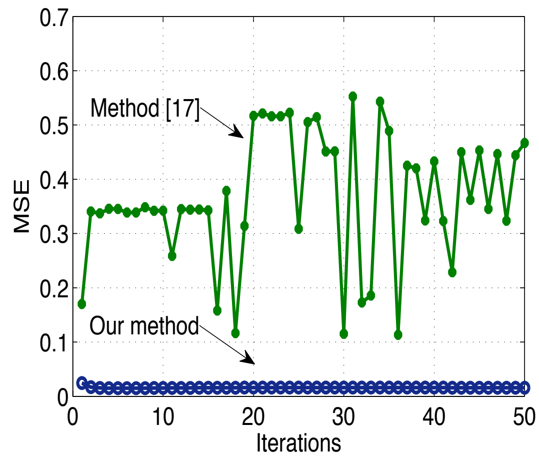


(d) Example 3: $p = 2$, $m = n_j = 100$, $s = 250$, $r_j = 20$, $\sigma_j = 1$, for $j = 1, 2$.

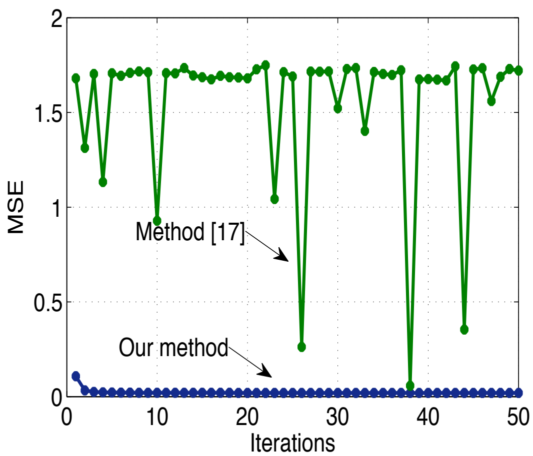
Fig. 3. Diagrams of MSE's associated with the proposed method and the known methods versus number of iterations, for $p = 2$ (i.e. for two sensors) and $p = 3$ (i.e. for three sensors), and different choices of signal dimensions, m, n_j, r_j, sample size s and noise 'level' σ_j.



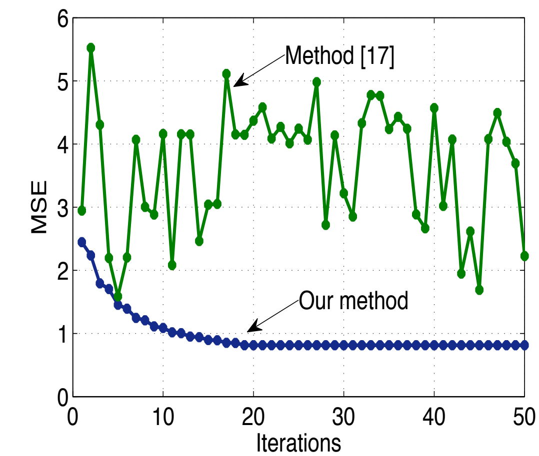
(a) Example 3: $p = 3$, $m = n_j = 100$, $s = 400$, $r_j = 20$, $\sigma_j = 1$, for $j = 1, 2, 3$.



(b) Example 4: $p = 3$, $m = n_j = 20$, $s = 20$, $r_j = 5$, $\sigma_j = 0.1j$, $j = 1, 2, 3$.

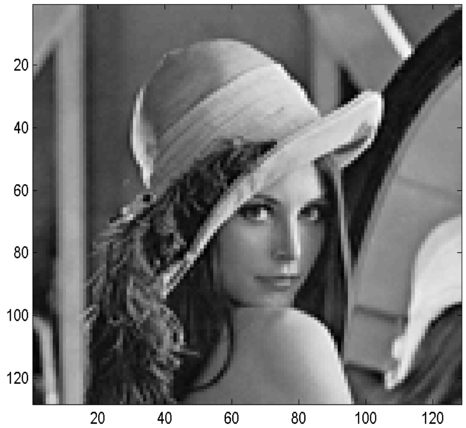


(c) Example 4: $p = 3$, $m = n_j = 100$, $s = 400$, $r_j = 25$, $\sigma_j = 0.1j$, for $j = 1, 2, 3$.

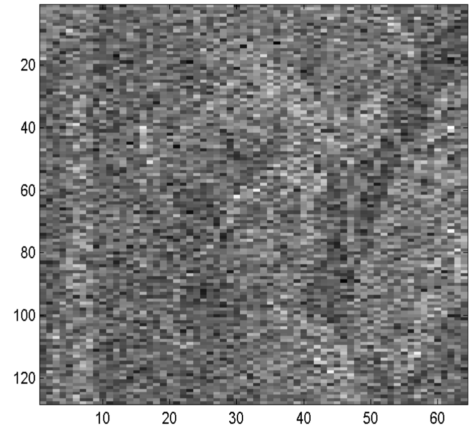


(d) Example 4: $p = 2$, $m = n_j = 100$, $s = 50$, $r_j = 20$, $\sigma_j = 0.1j$, for $j = 1, 2$.

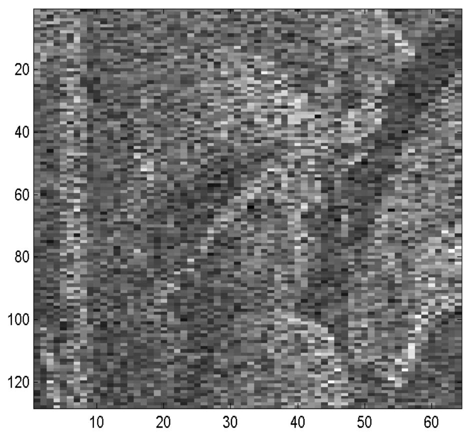
Fig. 4. Diagrams of MSE's associated with the proposed method and the known methods versus number of iterations, for $p = 2$ (i.e. for two sensors) and $p = 3$ (i.e. for three sensors), and different choices of signal dimensions, m, n_j, r_j, sample size s and noise 'level' σ_j.



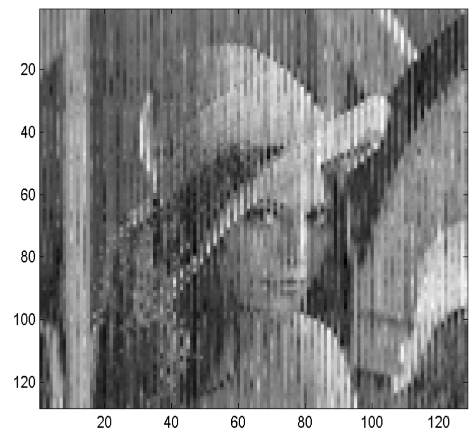
(a) Training reference signal X.



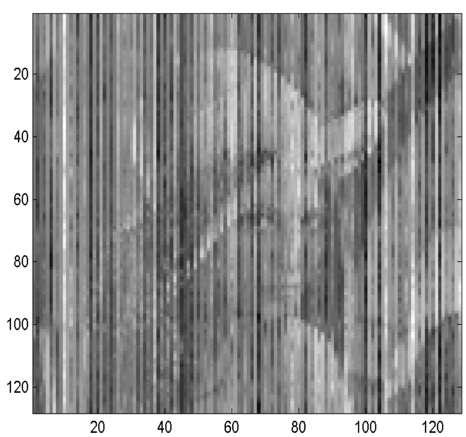
(b) Observed signal Y_1.



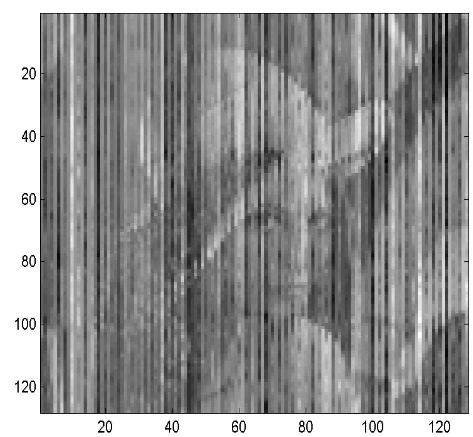
(c) Observed signal Y_2.



(d) Estimate of X by our method.

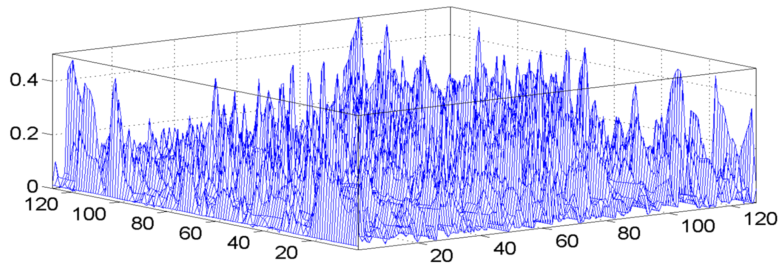


(e) Estimate of X by method [17].

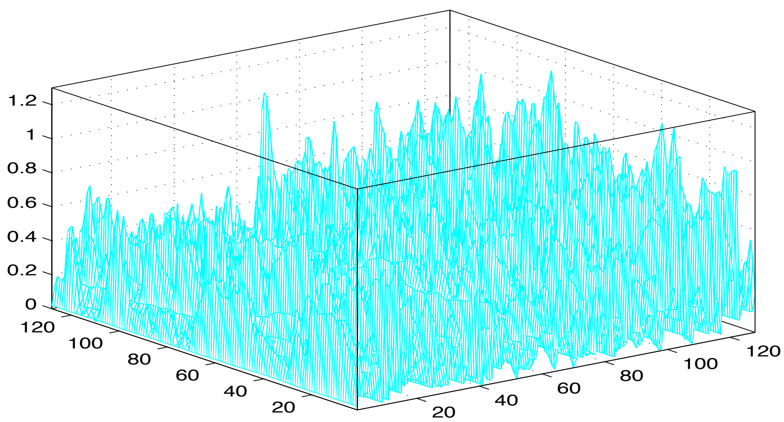


(f) Estimate of X by method [24].

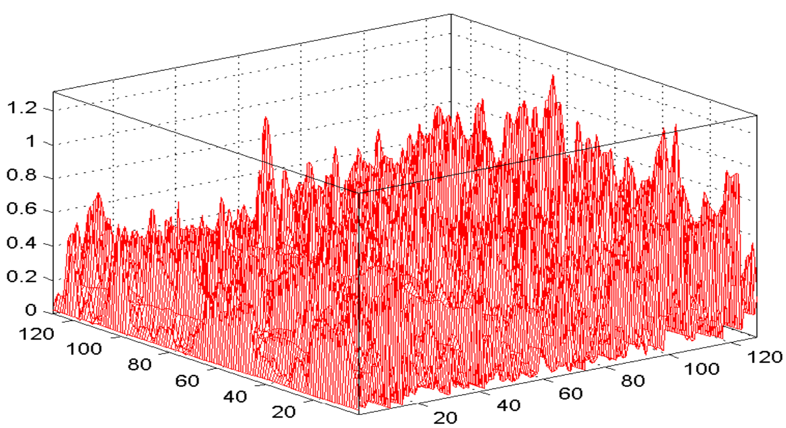
Fig. 5. Illustration to Example 5.



(a) Error associated with our method.



(b) Error associated with method [17].



(c) Error associated with method [24].

Fig. 6. Illustration to Example 5.